第五章思考题

2.4 Random用于产生随机数

2.5 字符串是字符的序列。一类是创建之后不会再做修复和变动的字符串，用string表示；另一类是创建之后允许再做更改和变化的字符串，用stringbuilder表示。

2.6 string类的concat方法的拼接速度远小于striingbuffer类的append方法。Concat方法是一个数组的拷贝，速度快，但最后的return语句创建一个新的string对象，也就是每次concat操作会创建一个新的string对象。

Append方法在做字符数组的处理，加长，拷贝等，这些是基础的数据处理，整个方法内没有生成对象，只是最后tostring返回一个对象。Append方法返回的是一个stringbuilder(or stringbuffer)的对象。

第五章编程题

3.2

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace \_2

{

class Program

{

static void Main(string[] args)

{

Console.WriteLine("请输入一个字符串");

string s = Console.ReadLine();

Console.WriteLine("请输入一个字符");

char x = Convert.ToChar(Console.ReadLine());

for (int i = 0; i < s.Length ; i++)

{

if (s[i] == x)

s.Remove(i);

else

Console.Write(s[i]);

}

}

}

}

3.3

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace \_3

{

class Program

{

static void Main(string[] args)

{

Console.WriteLine("请输入一个数字");

string s= Console.ReadLine();

string s1 = s;

if (s.Length % 2 == 0)

{

for (int i = 0; i < s.Length/2; i++)

{

if (s1[i] == s[s.Length - 1 - i])

{

if (i == s.Length / 2 - 1)

{

Console.WriteLine("{0}是回文数",s);

}

}

else

{

Console.WriteLine("{0}不是回文数",s);

}

}

}

else

{

for (int i = 0; i < s.Length - 1 / 2; i++)

{

if (s1[i] == s[s.Length - 1 - i])

{

if (i == s.Length - 1 / 2 - 1)

{

Console.WriteLine("{0}是回文数", s);

}

}

else

{

Console.WriteLine("{0}不是回文数", s);

}

}

}

}

}

}

3.13

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace \_13

{

class Program

{

static void Main(string[] args)

{

Console.WriteLine("输入十个数字");

string s = Console.ReadLine();

string[] s1 = s.Split(' ');

string[] s2 = s1;

string t;

for (int i = 0; i < s1.Length; i++)

{

Console.Write(s1[i]);

if (i == s1.Length - 1)

Console.WriteLine(' ');

}

for (int i = 0; i < s2.Length/2; i++)

{

t = s2[s2.Length - 1 - i];

s2[s2.Length - 1 - i] = s2[i];

s2[i] = t;

}

for (int i = 0; i < s2.Length; i++)

{

Console.Write(s2[i]);

if (i == s2.Length - 1)

Console.WriteLine(' ');

}

}

}

}